
LIBRARY

```
LL              IIIIII    BBBBBBBB      IIIIII    NN      NN    SSSSSSSS    QQQQQQ    TTTTTTTTTT    IIIIII
LL              IIIIII    BBBBBBBB      IIIIII    NN      NN    SSSSSSSS    QQQQQQ    TTTTTTTTTT    IIIIII
LL                II      BB      BB      II      NN      NN    SS        QQ      QQ      TT        II
LL                II      BB      BB      II      NN      NN    SS        QQ      QQ      TT        II
LL                II      BB      BB      II      NNNN    NN    SS        QQ      QQ      TT        II
LL                II      BB      BB      II      NNNN    NN    SS        QQ      QQ      TT        II
LL                II      BBBBBBBB        II      NN  NN  NN      SSSSSS  QQ      QQ      TT        II
LL                II      BBBBBBBB        II      NN  NN  NN      SSSSSS  QQ      QQ      TT        II
LL                II      BB      BB      II      NN    NNNN            SS  QQ  QQ  QQ      TT        II
LL                II      BB      BB      II      NN    NNNN            SS  QQ  QQ  QQ      TT        II
LL                II      BB      BB      II      NN      NN            SS  QQ    QQ        TT        II
LL                II      BB      BB      II      NN      NN            SS  QQ    QQ        TT        II
LLLLLLLLLL      IIIIII    BBBBBBBB      IIIIII    NN      NN    SSSSSSSS      QQQQ  QQ      TT      IIIIII
LLLLLLLLLL      IIIIII    BBBBBBBB      IIIIII    NN      NN    SSSSSSSS      QQQQ  QQ      TT      IIIIII
```

```
LL              IIIIII      SSSSSSSS
LL              IIIIII      SSSSSSSS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II        SSSSSS
LL                II        SSSSSS
LL                II              SS
LL                II              SS
LL                II              SS
LL                II              SS
LLLLLLLLLL      IIIIII    SSSSSSSS
LLLLLLLLLL      IIIIII    SSSSSSSS
```

Table of contents

```
                                                I  1
0000     1          .TITLE  LIB$INSQTI - Insert Entry into Queue at Tail, Interlocked
0000     2          .IDENT  /1-002/                 ; File: LIBINSQTI.MAR Edit:DGP1002
0000     3
0000     4 ;
0000     5 ;*****************************************************************************
0000     6 ;*                                                                           *
0000     7 ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000     8 ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     9 ;*  ALL RIGHTS RESERVED.                                                     *
0000    10 ;*                                                                           *
0000    11 ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0000    12 ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    13 ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0000    14 ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0000    15 ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0000    16 ;*  TRANSFERRED.                                                             *
0000    17 ;*                                                                           *
0000    18 ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0000    19 ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000    20 ;*  CORPORATION.                                                             *
0000    21 ;*                                                                           *
0000    22 ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0000    23 ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000    24 ;*                                                                           *
0000    25 ;*                                                                           *
0000    26 ;*****************************************************************************
0000    27 ;
0000    28
0000    29 ;++
0000    30 ; FACILITY: General Utility Library
0000    31 ;
0000    32 ; ABSTRACT:
0000    33 ;
0000    34 ;       One of four procedures which give higher level languages access
0000    35 ;       to the interlocked, self-relative queue instructions on the
0000    36 ;       VAX-11/780 and all future machines. This library procedure permits
0000    37 ;       the high level language user to have access to the INSQTI instruction.
0000    38 ;
0000    39 ; ENVIRONMENT: User Mode, AST Reentrant
0000    40 ;
0000    41 ;--
0000    42 ; AUTHOR: R. E. Johnston, CREATION DATE: 03-Dec-79
0000    43 ;
0000    44 ; MODIFIED BY:
0000    45 ;
0000    46 ;  1-001 - Original.  REJ 03-Dec-79
0000    47 ; 1-002 - Retry count is off by one.  DGP 14-AUG-1981
0000    48 ;--
```

```
                     0000     50          .SBTTL  DECLARATIONS
                     0000     51 ;
                     0000     52 ; INCLUDE FILES:
                     0000     53 ;
                     0000     54
                     0000     55 ;
                     0000     56 ; EXTERNAL DECLARATIONS:
                     0000     57 ;
                     0000     58          .DSABL  GBL                     ; Disable automatic generation
                     0000     59                                          ; of .EXTRN
                     0000     60          .EXTRN  SS$_NORMAL              ; Normal successful completion
                     0000     61          .EXTRN  LIB$_SECINTFAI          ; Secondary Interlock still
                     0000     62                                          ; locked after retry-cnt retrys
                     0000     63          .EXTRN  LIB$_ONEENTQUE          ; Successful Completion but
                     0000     64                                          ; only one entry is now
                     0000     65                                          ; contained in queue
                     0000     66 ;
                     0000     67 ; MACROS:
                     0000     68 ;
                     0000     69
                     0000     70 ;
                     0000     71 ; EQUATED SYMBOLS:
                     0000     72 ;
                0000000A     0000     73          DEF_RETRY_CNT = 10          ; Default retry count for
                     0000     74                                          ; Secondary Interlock fails
                     0000     75 ;
                     0000     76 ; OWN STORAGE:
                     0000     77 ;
                     0000     78
                     0000     79 ;
                     0000     80 ; PSECT DECLARATIONS:
                     0000     81 ;
                00000000     82          .PSECT _LIB$CODE PIC, SHR, LONG, EXE, NOWRT
                     0000     83
                     0000     84
                     0000     85
```

```
                        .SBTTL LIB$INSQTI - Insert Entry into Queue Tail
                ;++
                ; FUNCTIONAL DESCRIPTION:
                ;
                ;       One of four procedures which give higher level languages access
                ;       to the interlocked, self-relative queue instructions on the
                ;       VAX-11/780 and all future machines. This library procedure permits
                ;       the high level language user to have access to the INSQTI instruction.
                ;       With this procedure the user may insert a queue entry at the tail of a
                ;       user specified queue.
                ;
                ;       If the entry is successfully added to the tail of the queue and the
                ;       queue contains more than one entry, a successful completion status is
                ;       returned. If the entry is added to the tail of the queue and no other
                ;       entries are in the queue, the execution is successful but a unique
                ;       status value is returned indicating that the queue now contains one
                ;       entry (LIB$_ONEENTQUE).
                ;
                ;       These queue instructions are synchronized across all processors
                ;       through the use of a secondary interlock. The user may specify a
                ;       secondary interlock retry count. (The default retry count is 10.)
                ;       If the secondary interlock remains locked through retry-count retrys,
                ;       a secondary interlock status is returned to the user (LIB$_SECINTFAI)
                ;       and the entry is NOT successfully added to the head of the queue.
                ;
                ; CALLING SEQUENCE:
                ;
                ;   ret-status.wlc.v = LIB$INSQTI (entry.ml.ra, header.mq.r[, retry-cnt.rlu.r])
                ;
                ; INPUT PARAMETERS:
                ;
00000004        ;       ENTRY = 4                       ; Address of queue entry to be inserted
00000008        ;       HEADER = 8                      ; Address of queue header
0000000C        ;       RETRY_CNT = 12                  ; Address of retry count
                ;
                ; IMPLICIT INPUTS:
                ;
                ;       NONE
                ;
                ; OUTPUT PARAMETERS:
                ;
                ;       NONE
                ;
                ; IMPLICIT OUTPUTS:
                ;
                ;       NONE
                ;
                ; FUNCTION VALUE:
                ;
                ;       SS$_NORMAL      - Entry added to head of queue, queue contains more
                ;                         than one entry.
                ;       LIB$_ONEENTQUE  - Successful completion of instruction (INSQTI).
                ;                         Entry added to tail of queue, but queue was empty.
                ;       LIB$_SECINTFAI  - Secondary Interlock failed, queue is not modified.
                ;
```

```
                                0000    144 ; SIDE EFFECTS:
                                0000    145 ;
                                0000    146 ;       SS$_ROPRAND - reserved operand fault for:
                                0000    147 ;                     1.) either the entry or the header is at an address
                                0000    148 ;                         that is not quad word aligned.
                                0000    149 ;                     2.) address of header equals address of entry.
                                0000    150 ;
                                0000    151 ;--
                                0000    152
                           0000 0000    153         .ENTRY  LIB$INSQTI , ^M< >      ; Entry point
                                0002    154
                      50   0A   D0 0002    155         MOVL    #DEF_RETRY_CNT, R0      ; R0 = Default retry count of 10
                      03   6C   91 0005    156         CMPB    (AP), #<RETRY_CNT/4>    ; Check for optional retry cnt operand
                           04   1F 0008    157         BLSSU   20$                     ; Branch if default count to be used
                 50    0C BC   D0 000A    158         MOVL    @RETRY_CNT(AP), R0      ; R0 = User specified retry count
                                000E    159 20$:
              08 BC    04 BC   5D 000E    160         INSQTI  @ENTRY(AP), @HEADER(AP) ; Do the instruction (INSQTI)
                           08   1B 0013    161         BLEQU   30$                     ; Branch if Z = 1 (One entry in queue)
                                0015    162                                         ; or C = 1 (Secondary Interlock fail)
           50   00000000'8F   D0 0015    163         MOVL    #SS$_NORMAL, R0         ; Normal status - Entry added to tail
                                001C    164                                         ; of queue and more than one entry is
                                001C    165                                         ; now is queue
                           04   001C    166         RET                             ; Successful return to user
                                001D    167 30$:
                           08   1F 001D    168         BCS     40$                     ; Branch if Secondary Interlock fail
           50   00000000'8F   D0 001F    169         MOVL    #LIB$_ONEENTQUE, R0     ; Assume exactly one entry now in queue
                                0026    170                                         ; Entry successfully entered into queue
                           04   0026    171         RET                             ; Successful return to user
                                0027    172 40$:
                      E4 50     F4 0027    173         SOBGEQ  R0, 20$                 ; Loop until retry count is exhausted
           50   00000000'8F   D0 002A    174         MOVL    #LIB$_SECINTFAI, R0     ; Retry count is exhausted
                                0031    175                                         ; Secondary Interlock fail status
                           04   0031    176         RET                             ; Unsuccessful return to user
                                0032    177         .END
```

```
DEF_RETRY_CNT   = 0000000A
ENTRY           = 00000004
HEADER          = 00000008
LIB$INSQTI        00000000 RG    01
LIB$_ONEENTQUE    ********   X   00
LIB$_SECINTFAI    ********   X   00
RETRY_CNT       = 0000000C
SS$_NORMAL        ********   X   00
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
_LIB$CODE	00000032 (50.)	01 (1.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC LONG

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	30	00:00:00.06	00:00:01.69
Command processing	104	00:00:00.28	00:00:02.76
Pass 1	67	00:00:00.29	00:00:03.07
Symbol table sort	0	00:00:00.00	00:00:00.01
Pass 2	47	00:00:00.20	00:00:02.30
Symbol table output	2	00:00:00.01	00:00:00.01
Psect synopsis output	2	00:00:00.01	00:00:00.24
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	254	00:00:00.86	00:00:10.08

The working set limit was 900 pages.
1848 bytes (4 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 8 non-local and 3 local symbols.
177 source lines were read in Pass 1, producing 11 object records in Pass 2.
0 pages of virtual memory were used to define 0 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYSLIB]STARLET.MLB;2	0

0 GETS were required to define 0 macros.

There were no errors, warnings or information messages.

```
MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS$:LIBINSQTI/OBJ=OBJ$:LIBINSQTI MSRC$:LIBINSQTI/UPDATE=(ENH$:LIBINSQTI)
```

LIBLOCC
LIS

LIBLUN
LIS

LIBMOVC3
LIS

LIBMOVTC
LIS

LIBPOLYF
LIS

LIBINSQTI
LIS

LIBINTOVE
LIS

LIBLEXICA
LIS

LIBPKARIT
LIS

LIBMATCHC
LIS

LIBPOLYD
LIS

LIBLOOKUP
LIS

LIBLPLINE
LIS

LIBMOVC5
LIS

LIBLEN
LIS

LIBMSG
LIS

LIBINSV
LIS

LIBMATCH
LIS

LIBMOVTUC
LIS